CLAIMS

What is claimed is:

1. A data modulating method of modulating m-bit source data into n-bit, wherein $n \geq m$, and m and n are positive integers, code words where a minimum run length limit is confined to “d” and a maximum run length limit is confined to “k”, wherein "d" and "k" are positive integers, the data modulating method comprising:

dividing an input data stream by a predetermined length and multiplexing the input data stream using multiplexed information according to a predetermined multiplexing method to provide a multiplexed data stream;

weak DC-free run length limited (RLL)-modulating the multiplexed data stream without a separate DC control code conversion table comprising additional bits; and

providing a code stream having a minimum DC component among multiplexed, RLL-modulated code streams.

2. The data modulating method of claim 1, wherein the multiplexing method uses a scrambling method.

3. The data modulating method of claim 1, wherein the multiplexing method uses an interleaving method.

4. The data modulating method of claim 1, wherein the weak DC-free RLL modulation is performed by generating code words complying with a predetermined RLL condition and grouping the code words according to the predetermined RLL condition using a main code conversion table in which code words are arranged so that a stream of source words has a DC control capability and a DC control sub code conversion table which is made by taking unnecessary code words complying with the predetermined RLL condition out of the main code conversion table.

5. The data modulating method of claim 4, wherein the minimum run length limit “d” is 1 and the maximum run length limit “k” is 7.

6. The data modulating method of claim 4, wherein in the weak DC-free RLL modulation, code words of same data in main code groups and in DC control auxiliary code groups, have opposite INV values to control a direct current (DC), wherein INV is a parameter predicting a transition direction of a next code word depending on whether a number of bits of value "1" in a code word stream is odd or even.

7. The data modulating method of claim 4, wherein the main code conversion table comprises code word groups that satisfy conditions of $0 \le EZ \le 5$, $1 \le LZ \le 7$, $0 \le LZ \le 4$, and $0 \le LZ \le 2$, wherein EZ is end zeros indicating a number of successive zeros from least significant bits to most significant bits in a code word and LZ is lead zeros indicating a number of successive zeros from most significant bits to least significant bits.

8. The data modulating method of claim 7, wherein one of the code groups of the main code conversion table comprising source codes less than a minimum number of code words to be modulated is prepared by removing redundant code words from one of the code groups comprising code words greater than the minimum number so that code words greater than the minimum number are retained.

9. The data modulating method of claim 4, wherein the DC control sub conversion table comprises:

a group comprising code words satisfying $6 \le EZ \le 7$ and $LZ \ne 0$, redundant code words of a first main code group, and "1010xxxxxxxx" code words satisfying $6 \le EZ \le 7$ and $LZ = 0$, wherein EZ is end zeros indicating a number of successive zeros from most significant bits to least significant bits and LZ is lead zeros indicating a number of successive zeros from most significant bits to least significant bits;

a group comprising code words satisfying $6 \le EZ \le 7$ and $0 \le LZ \le 6$, code words satisfying $0 \le EZ \le 5$ and $5 \le LZ \le 6$, and redundant code words of a second main code group, the group from which "1010xxxxxxxx" code words satisfying $6 \le EZ \le 7$, $LZ = 0$ are removed; and

a group comprising code words complying with $6 \le EZ \le 7$ and $0 \le LZ \le 3$, code words satisfying $0 \le EZ \le 5$ and $LZ = 3$, and redundant code words of a third main code group.

10. The data modulating method of claim 9, wherein when a stream of code words **a**, b1, and c, and a stream of code words **a**, b2, and c make a pair and the code words b1 and b2 are code words having opposite INV characteristics in the DC control sub code conversion table, although the code word **a**, b1, b2, or **c** is converted due to a violation against a predetermined run length limit (boundary rule) between the stream of the code words **a**, b1, and b2 and the stream of the code words b1, b2, and **c**, code words are arranged in the main code conversion table and the DC control sub code conversion table so that a modulated stream of code words **a**, b1, and **c** and a modulated stream of code words **a**, b2, and **c** have opposite INV characteristics.

11. The data modulating method of claim 4, wherein the minimum run length limit "d" is 2 and the maximum run length limit "k" is 10.

12. The data modulating method of claim 1, further comprising inserting a sync pattern into the multiplexed data stream to which the multiplexed information is added and converting the multiplexed information into multiplexed IDs.

13. The data modulating method of claim 12, wherein the multiplexed IDs satisfy the minimum run length limit "d" of 2 and the maximum run length limit "k" of 7 to increase a minimum mark length to reduce interferential noise of a signal.

14. A data modulating method of modulating m-bit source data into n-bit, wherein $n \geq m$, and m and n are positive integers, code words where a minimum run length limit is confined to "d" and a maximum run length limit is confined to "k", wherein "d" and "k" are positive integers, the data modulating method comprising:

multiplexing an input data stream divided by a predetermined length into a plurality of types of pseudo random data streams using multiplexed information of predetermined bits by applying a predetermined multiplexing method to each of the pseudo random data streams; and

RLL-modulating the plurality of types of pseudo random data streams to create a modulated code stream in which an optimal DC suppression is performed.

15. The data modulating method of claim 14, wherein the random data streams are generated by inconsecutively scrambling the input data stream using the multiplexed information.

16. The data modulating method of claim 15, wherein when s_t is multiplexed information used to multiplex the input data stream divided into **v** data streams each having a u-byte length, a number **a** of bits of the multiplexed information s_t is less than or equal to a number **m** of bits of input source data, wherein **u** and **v** are positive integers.

17. The data modulating method of claim 16, wherein the generation of random data streams comprises:

performing an exclusive OR operation on the multiplexed information s_t and first m-bit data of a first code modulation unit of the plurality of types of random data streams to generate modulated data;

outputting data unchanged from a second code modulation unit to a $q-1^{th}$ code modulation unit without performing an exclusive OR operation;

performing an exclusive OR operation on the first code modulation unit and a q^{th} code modulation unit to generate next modulated data; and

repeating an exclusive OR operation from the q^{th} code modulation unit to a final code modulation unit of the input data stream.

18. The data modulating method of claim 17, wherein when an exclusive OR operation cycle is **q**, where **q** is a positive integer, an error propagation probability is reduced to 1/q.

19. The data modulating method of claim 14, further comprising:

dividing the input data stream into the predetermined length;

inserting a sync pattern into the multiplexed pseudo random data streams to which the multiplexed information is added in the generation of the random data streams and converting the multiplexed information into multiplexed IDs; and

comparing the plurality of types of RLL-modulated code streams to select the code stream comprising the minimum of DC components.

20. A data modulating method of modulating m-bit source data into n-bit, wherein n ≥ m, and m and n are positive integers, code words where a minimum run length limit is confined to “d” and a maximum run length limit is confined to “k”, wherein "d" and "k" are positive integers, the data modulating method comprising:

multiplexing an input data stream divided by a predetermined length into a plurality of types of pseudo random data streams using multiplexed information of predetermined bits by applying a predetermined multiplexing method to each of the pseudo random data streams; and

weak DC-free RLL-modulating the multiplexed data streams without using a DC control code conversion table comprising additional bits and providing a code stream comprising a minimum of DC components among multiplexed, RLL-modulated code streams.

21. The data modulating method of claim 20, wherein the pseudo random data streams are generated by inconsecutively scrambling the input data stream using the multiplexed information.

22. The data modulating method of claim 21, wherein s_t is multiplexed information used to multiplex the input data stream divided into **v** data streams each having a u-byte length, a number **a** of bits of the multiplexed information s_t is less than or equal to a number **m** of bits of input source data, wherein **u** and **v** are integers.

23. The data modulating method of claim 22, wherein the generation of the pseudo random data streams comprises:

performing an exclusive OR operation on the multiplexed information s_t and first m-bit data of a first code modulation unit of the plurality of types of pseudo random data streams to generate modulated data;

outputting data unchanged from a second code modulation unit to a $q-1^{th}$ code modulation unit without performing an exclusive OR operation;

performing an exclusive OR operation on the first code modulation unit and a q^{th} code modulation unit to generate next modulated data; and

repeating an exclusive OR operation from the q^{th} code modulation unit to a final code modulation unit of the input data stream.

24. The data modulating method of claim 23, wherein when an exclusive OR operation cycle is **q**, where **q** is a positive integer, an error propagation probability is reduced to 1/q.

25. The data modulating method of claim 20, further comprising:

dividing the input data stream into the predetermined length;

inserting a sync pattern into the multiplexed pseudo random data streams to which the multiplexed information is added in generation of the random data streams and converting the multiplexed information into multiplexed IDs; and

comparing the plurality of types of RLL-modulated code streams to select the code stream comprising the minimum of DC components.

26. The data modulating method of claim 20, wherein the weak DC-free RLL modulation is performed by generating code words complying with a predetermined RLL condition and grouping the code words according to the predetermined RLL condition using a main code conversion table in which code words are arranged so that a stream of source words has a DC control capability and using a DC control sub code conversion table which is made by taking unnecessary code words complying with the predetermined RLL condition out of the main code conversion table.

27. The data modulating method of claim 26, wherein the minimum run length limit "d" is 1 and the maximum run length limit "k" is 7.

28. The data modulating method of claim 26, wherein in the weak DC-free RLL modulation, code words of same data in main code groups and in DC control auxiliary code groups, have opposite INV values to control a DC, and INV is a parameter predicting a transition direction of a next code word depending on whether a number of bits of value "1" in a code word stream is odd or even.

29. The data modulating method of claim 26, wherein the main code conversion table comprises code word groups that satisfy conditions of $0 \le EZ \le 5$, $1 \le LZ \le 7$, $0 \le LZ \le 4$, and $0 \le LZ \le 2$, wherein EZ is end zeros indicating a number of successive zeros from least significant bits to most significant bits in a code word and LZ is lead zeros indicating a number of successive zeros from most significant bits to least significant bits.

30. The data modulating method of claim 29, wherein one of the code groups of the main code conversion table comprising source codes less than a minimum number of code words to be modulated is prepared by removing redundant code words from one of the code groups comprising code words greater than the minimum number so that code words greater than the minimum number are retained.

31. The data modulating method of claim 26, wherein the DC control sub conversion table comprises:

a group comprising code words satisfying $6 \le EZ \le 7$ and $LZ \neq 0$, redundant code words of a first main code group, and "1010xxxxxxxx" code words satisfying $6 \le EZ \le 7$ and $LZ = 0$;

a group comprising code words satisfying $6 \le EZ \le 7$ and $0 \le LZ \le 6$, code words satisfying $0 \le EZ \le 5$ and $5 \le LZ \le 6$, and redundant code words of a second main code group, the group from which "1010xxxxxxxx" code words satisfying $6 \le EZ \le 7$, $LZ = 0$ are removed, wherein EZ is end zeros indicating a number of successive zeros from most significant bits to least significant bits and LZ is lead zeros indicating a number of successive zeros from most significant bits to least significant bits; and

a group comprising code words complying with $6 \le EZ \le 7$ and $0 \le LZ \le 3$, code words satisfying $0 \le EZ \le 5$ and $LZ = 3$, and redundant code words of a third main code group.

32. The data modulating method of claim 31, wherein when a stream of code words **a**, b1, and c, and a stream of code words **a**, b2, and c make a pair and the code words b1 and b2 are code words having opposite INV characteristics in the DC control sub code conversion table, although the code word **a**, b1, b2, or **c** is converted due to a violation against a predetermined run length limit/boundary rule between the stream of the code words **a**, b1, and b2 and the stream of the code words b1, b2, and **c**, code words are arranged in the main code conversion table and the DC control sub code conversion table so that a modulated stream of code words **a**, b1, and **c** and a modulated stream of code words **a**, b2, and **c** have opposite INV characteristics.

33. The data modulating method of claim 26, wherein the minimum run length limit "d" is 2 and the maximum run length limit "k" is 10.

34. The data modulating method of claim 25, wherein the multiplexed IDs satisfy the minimum run length limit "d" of 2 and the maximum run length limit "k" of 7 to increase a minimum mark length to reduce interferential noise of a signal.

35. A data demodulating method comprising:

demodulating each **n** bit of input digital data into m-bit where n ≥ m, and m and n are positive integers, of a demodulation code unit to generate a non-inverted data stream having a predetermined length; and

inconsecutively descrambling the non-inverted data stream using multiplexed information to generate an inverted data stream.

36. The demodulating method of claim 35, wherein the descrambling of the non-inverted data stream comprises:

performing an exclusive OR operation on a first demodulation code unit and initial data that is the multiplexed information to generate first inverted data;

outputting a second demodulation code unit to a q-1th demodulation code unit without performing an exclusive OR operation;

performing an exclusive OR operation on the first demodulation code unit and a first qth demodulation code unit of inconsecutive qth demodulation code units to generate next inverted data; and

repeating an exclusive OR operation to the final one of the remaining qth demodulation code units of the non-inverted data stream to provide the inverted data stream.

37. A method of arranging m-bit source data into n-bit, where n ≥ m, and n and m are positive integers, code words by confining a minimum run length limit “d” to 1 and a maximum run length limit “k” to 7, where "d" and "k" are positive integers, the method comprising:

when a code word **a** is connected to a code word **b**, the code word **a** is a preceding code word, the code word b is selected from code words b1 and b2, a code stream in which the code word **a** is connected to the code word b1 is X1, and a code stream in which the code word **a** is connected to the code word b2 is X2, arranging the code words b1 and b2 to have opposite parameters INV predicting a transition of a next code depending on whether a number of bits of value “1” in a code word is odd or even; and

when the code word **a** is connected to the code word b1 or b2, although the code word **a**, b1, or b2 is modulated into another type of code word according to a boundary rule, arranging the code streams X1 and X2 to have opposite parameters INV.

38. The method of claim 37, wherein when a number of bits of value “0” between the code word **a** and the code word b1 and between the code word **a** and the code word b2 is less than the minimum run length limit “d"=1, the code word **a**, b1, or b2 is modified according to the boundary rule to arrange the code words **a**, b1, and b2 so that a number of bits of value “0” between the modified code word **a** and the modified code word b1 and between the modified code word **a** and the modified code word b2 is greater than or equal to the minimum run length limit "d"=1 and less than or equal to the maximum run length limit "k"=7.

39. The method of claim 37, wherein the code word **a** in the code stream X1 and the code word **a** in the code stream X2 are each converted into another type of code word according to the boundary rule to have the same INV value so that the code streams X1 and X2 have opposite INV values due to INV values of the code words b1 and b2.

40. A method of arranging m-bit source data into n-bit, wherein n ≥ m, and n and m are positive integers, code words by confining a minimum run length limit “d” to 1 and a maximum run length limit “k” to 7, where "d" and "k" are positive integers, the method comprising:

when a code word **b** is connected to a code word **c**, the code word **b** is a preceding code word, the code word **b** is selected from code words b1 and b2, a code stream in which the code word b1 is connected to the code word **c** is Y1, and a code stream in which the code word b2 is connected to the code word **c** is Y2, the code words b1 and b2 are arranged to have opposite parameters INV predicting a transition of a next code depending on whether a number of bits of value “1” in a code word is odd or even; and

when the code word b1 or b2 is connected to the code word **c**, although the code word b1, b2, or **c** is modulated into another type of code word according to a boundary rule, arranging the code streams Y1 and Y2 to have opposite parameters INV.

41. An apparatus for modulating m-bit source data into n-bit, wherein n ≥ m, and n and m are positive integers, code words to improve a DC suppression capability by confining a minimum run length limit to “d” and a maximum run length limit to “k”, where "d" and "k" are positive integers, the apparatus comprising:

a multiplexer that multiplexes input data divided by a predetermined length using multiplexed information to provide a multiplexed data stream;

an encoder that weak DC-free RLL-modulates the multiplexed data stream without using a DC control sub code conversion table to which additional bits are added; and

a selector that selects a code stream comprising a minimum DC component among multiplexed, RLL-modulated code streams.

42. The apparatus of claim 41, further comprising:

a divider that divides the input data stream into the predetermined length; and

a sync signal and multiplexed ID inserter that inserts a sync pattern into the multiplexed data stream to which the multiplexed information is added and converts the multiplexed information into multiplexed IDs.

43. The apparatus of claim 41, wherein the multiplexer converts the divided input data stream into random data using a scrambling method.

44. The apparatus of claim 41, wherein the multiplexer converts the divided input data stream into random data using an interleaving method.

45. The apparatus of claim 41, wherein the encoder performs the weak DC-free RLL modulation using code words of the same data in main code groups and in DC control auxiliary code groups, the code words having opposite INV values to control a DC_1 and INV being a parameter predicting a transition direction of a next code word depending on whether a number of bits of value "1" in a code word stream is odd or even.

46. The apparatus of claim 41, wherein the encoder performs the weak DC-free RLL modulation by confining the minimum run length limit "d" to 1 and the maximum run length limit "d" to 7.

47. The apparatus of claim 46, wherein when a bit length of the source data is 8, the modulated code words have a 12-bit length.

48. The apparatus of claim 42, wherein the sync signal and multiplexed ID inserter performs the inserting and multiplexing by confining the minimum run length limit "d" to 1 and the maximum run length limit "k" to 7.

49. The apparatus of claim 42, wherein the encoder confines the minimum run length limit "d" to 1 and the maximum run length limit "k" to 7, while the sync signal and multiplexed ID inserter confines the minimum run length limit "d" to 2 and the maximum run length limit "k" to 7 to increase the minimum run length limit "d" so as to increase a minimum mark length so that interferential noise of a signal is reduced.

50. The apparatus of claim 42, wherein the encoder performs the weak DC-free RLL modulation by confining the minimum run length limit "d" to 2 and the maximum run length limit "d" to 10.

51. The apparatus of claim 50, wherein when a bit length of the source data is 8, the modulated code words have a 15-bit length.

52. The apparatus of claim 50, wherein the sync signal and multiplexed ID inserter performs the inserting and multiplexing by confining the minimum run length limit "d" to 2 and the maximum run length limit "k" to 10.

53. A data modulating apparatus to modulate m-bit source data into n-bit, wherein n ≥ m, and n and m are positive integers, code words where a minimum run length limit is confined to "d" and a maximum run length limit is confined to "k", wherein "d" and "k" are positive integers, the data modulating apparatus comprising:

a multiplexer that multiplexes an input data stream divided by a predetermined length into a plurality of types of pseudo random data streams using multiplexed information of predetermined bits by applying a predetermined multiplexing method to each of the pseudo random data streams; and

an encoder that RLL-modulates the plurality of types of pseudo random data streams to create a modulated code stream in which an optimal DC suppression is performed.

54. The apparatus of claim 53, wherein the multiplexer generates the random data streams by inconsecutively scrambling the input data stream using the multiplexed information.

55. The apparatus of claim 54, wherein when s_t is multiplexed information used to multiplex the input data stream divided into **v** data streams each having a u-byte length, a number **a** of bits of the multiplexed information s_t is less than or equal to a number **m** of bits of input source data, wherein **u** and **v** are positive integers.

56. The apparatus of claim 55, wherein the multiplexer comprises exclusive OR devices, each of which is disposed for each q^{th} code modulation unit of the plurality of types of random data streams, and performs an exclusive OR operation on the multiplexed information s_t and first m-bit data of a first code modulation unit of the plurality of types of random data streams using the first exclusive OR devices to generate modulated data, outputs data unchanged from a second code modulation unit to a $q\text{-}1^{th}$ code modulation unit without an exclusive OR operation, performs an exclusive OR operation on the first code modulation unit and a q^{th} code modulation unit to generate next modulated data, and repeats an exclusive OR operation from the q^{th} code modulation unit to a final code modulation unit of the input data stream.

57. The apparatus of claim 56, wherein when an exclusive OR operation cycle is **q**, where **q** is a positive integer, an error propagation probability is reduced to 1/q.

58. The apparatus of claim 53, further comprising:

a divider that divides the input data stream into the predetermined length;

a sync signal and multiplexed ID inserter that inserts a sync pattern into the multiplexed pseudo random data streams to which the multiplexed information is added and converts the multiplexed information into multiplexed IDs; and

a comparator and selector that compares the plurality of types of RLL-modulated code streams to select the code stream comprising the minimum of DC components.

59. A data modulating apparatus to modulate m-bit source data into n-bit, wherein n ≥ m, and n and m are positive integers, code words where a minimum run length limit is confined to “d” and a maximum run length limit is confined to “k”, wherein "d" and "k" are positive integers, the data modulating apparatus comprising:

a multiplexer that multiplexes an input data stream divided by a predetermined length into a plurality of types of pseudo random data streams using multiplexed information of predetermined bits by applying a predetermined multiplexing method to each of the pseudo random data streams;

an encoder that weak DC-free RLL-modulates the multiplexed data stream without using DC control sub code conversion table to which additional bits are added; and

a selector that selects a code stream comprising a minimum of DC components among multiplexed, RLL-modulated code streams.

60. The apparatus of claim 59, wherein the multiplexer generates the random data streams by inconsecutively scrambling the input data stream using the multiplexed information.

61. The apparatus of claim 60, wherein when s_t is multiplexed information used to multiplex the input data stream divided into **v** data streams each having a u-byte length, a number **a** of bits of the multiplexed information s_t is less than or equal to a number **m** of bits of input source data, wherein **u** and **v** are positive integers.

62. The apparatus of claim 61, wherein the multiplexer comprises exclusive OR devices, each of which is disposed for each q^{th} code modulation unit of the plurality of types of random data streams, and performs an exclusive OR operation on the multiplexed information s_t and first m-bit data of a first code modulation unit of the plurality of types of random data streams using the first exclusive OR devices to generate modulated data, outputs a second code modulation unit to a $q\text{-}1^{th}$ code modulation unit without an exclusive OR operation, performs an exclusive OR operation on the first code modulation unit and a q^{th} code modulation unit to generate next modulated data, and repeats an exclusive OR operation from the q^{th} code modulation unit to a final code modulation unit of the input data stream.

63. The apparatus of claim 62, wherein when an exclusive OR operation cycle is **q**, where **q** is a positive integer, an error propagation probability is reduced to 1/q.

64. The apparatus of claim 59, further comprising:

a divider that divides the input data stream into the predetermined length; and

a sync signal and multiplexed ID inserter that inserts a sync pattern into the multiplexed data stream to which the multiplexed information is added and converts the multiplexed information into multiplexed IDs.

65. The apparatus of claim 59, wherein the encoder performs the weak DC-free RLL modulation when code words of the same data in main code groups and in DC control auxiliary code groups, have opposite INV values in order to control a DC, and INV is a parameter predicting a transition direction of a next code word depending on whether a number of bits of value "1" in a code word stream is odd or even.

66. The apparatus of claim 59, wherein the encoder performs the weak DC-free RLL modulation by confining the minimum run length limit "d" to 1 and the maximum run length limit "d" to 7.

67. The apparatus of claim 66, wherein when a bit length of the source data is 8, the modulated code words have a 12-bit length.

68. The apparatus of claim 64, wherein the sync signal and multiplexed ID inserter performs the inserting and multiplexing by confining the minimum run length limit "d" to 1 and the maximum run length limit "k" to 7.

69. The apparatus of claim 64, wherein the encoder confines the minimum run length limit "d" to 1 and the maximum run length limit "k" to 7, while the sync signal and multiplexed ID inserter confines the minimum run length limit "d" to 2 and the maximum run length limit "k" to 7 to increase the minimum run length limit "d" so as to increase a minimum mark length so that interferential noise of a signal is reduced.

70. The apparatus of claim 64, wherein the encoder performs the weak DC-free RLL modulation by confining the minimum run length limit "d" to 2 and the maximum run length limit "d" to 10.

71. The apparatus of claim 70, wherein when a bit length of the source data is 8, the modulated code words have a 15-bit length.

72. The apparatus of claim 70, wherein the sync signal and multiplexed ID inserter performs the inserting and multiplexing by confining the minimum run length limit "d" to 2 and the maximum run length limit "k" to 10.

73. A data demodulating apparatus comprising:

a decoder demodulating each **n** bit of input digital data into m-bit, wherein $n \geq m$, and n and m are positive integers, of a demodulation code unit to generate a non-inverted data stream having a predetermined length; and

a demultiplexer that inconsecutively descrambles the non-inverted data stream using multiplexed information to generate an inverted data stream.

74. The apparatus of claim 73, wherein the demultiplexer comprises exclusive OR devices, each of which is disposed for each q^{th} demodulation code unit and performs an exclusive OR operation on a first demodulation code unit and initial data that is the multiplexed information to generate first inverted data, outputs a second demodulation code unit to a $q-1^{th}$ demodulation code unit without an exclusive OR operation, performs an exclusive OR operation on the first demodulation code unit and a first q^{th} demodulation code unit of inconsecutive q^{th} demodulation code units to generate next inverted data, and repeats an exclusive OR operation to the final one of the remaining q^{th} demodulation code units of the non-inverted data stream to provide the inverted data stream.

75. A computer-readable medium having computer-executable instructions for performing operations of a data modulating method of modulating m-bit source data into n-bit, wherein $n \geq m$ and m and n are positive integers, code words where a minimum run length limit is confined to "d" and a maximum run length limit is confined to "k", wherein "d" and "k" are positive integers, the operations comprising:

dividing an input data stream by a predetermined length and multiplexing the input data stream using multiplexed information according to a predetermined multiplexing method to provide a multiplexed data stream;

weak DC-free run length limited (RLL)-modulating the multiplexed data stream without a separate DC control code conversion table comprising additional bits; and

providing a code stream having a minimum of DC components among multiplexed, RLL-modulated code streams.

76. A computer-readable medium having computer-executable instructions for performing operations of a data modulating method of modulating m-bit source data into n-bit, wherein n ≥ m and m and n are positive integers, code words where a minimum run length limit is confined to “d” and a maximum run length limit is confined to “k”, wherein "d" and "k" are positive integers, the operations comprising:

multiplexing an input data stream divided by a predetermined length into a plurality of types of pseudo random data streams using multiplexed information of predetermined bits by applying a predetermined multiplexing method to each of the pseudo random data streams; and

RLL-modulating the plurality of types of pseudo random data streams to create a modulated code stream in which an optimal DC suppression is performed.

77. A computer-readable medium having computer-executable instructions for performing operations of a data modulating method of modulating m-bit source data into n-bit, wherein n ≥ m and m and n are positive integers, code words where a minimum run length limit is confined to “d” and a maximum run length limit is confined to “k”, wherein "d" and "k" are positive integers, the operations comprising:

multiplexing an input data stream divided by a predetermined length into a plurality of types of pseudo random data streams using multiplexed information of predetermined bits by applying a predetermined multiplexing method to each of the pseudo random data streams; and

weak DC-free RLL-modulating the multiplexed data streams without using a DC control code conversion table comprising additional bits and providing a code stream comprising a minimum of DC components among multiplexed, RLL-modulated code streams.

78. A computer-readable medium having computer-executable instructions for performing operations of a data demodulating method, the operations comprising:

demodulating each **n** bit of input digital data into m-bit, where n ≥ m and m and n are positive integers, of a demodulation code unit to generate a non-inverted data stream having a predetermined length; and

inconsecutively descrambling the non-inverted data stream using multiplexed information to generate an inverted data stream.

79. A computer-readable medium having computer-executable instructions for performing operations of a method of arranging m-bit source data into n-bit, where n ≥ m and n and m are positive integers, code words by confining a minimum run length limit “d” to 1 and a maximum run length limit “k” to 7, where "d" and "k" are positive integers, the operations comprising:

when a code word **a** is connected to a code word **b**, the code word **a** is a preceding code word, the code word b is selected from code words b1 and b2, a code stream in which the code word **a** is connected to the code word b1 is X1, and a code stream in which the code word **a** is connected to the code word b2 is X2, arranging the code words b1 and b2 to have opposite parameters INV predicting a transition of a next code depending on whether a number of bits of value “1” in a code word is odd or even; and

when the code word **a** is connected to the code word b1 or b2, although the code word **a**, b1, or b2 is modulated into another type of code word according to a boundary rule, arranging the code streams X1 and X2 to have opposite parameters INV.

80. A computer-readable medium having computer-executable instructions for performing operations of a method of arranging m-bit source data into n-bit, wherein n ≥ m and n and m are positive integers, code words by confining a minimum run length limit “d” to 1 and a maximum run length limit “k” to 7, where "d" and "k" are positive integers, the operations comprising:

when a code word **b** is connected to a code word **c**, the code word **b** is a preceding code word, the code word **b** is selected from code words b1 and b2, a code stream in which the code word b1 is connected to the code word **c** is Y1, and a code stream in which the code word b2 is connected to the code word **c** is Y2, the code words b1 and b2 are arranged to have opposite parameters INV predicting a transition of a next code depending on whether a number of bits of value “1” in a code word is odd or even; and

when the code word b1 or b2 is connected to the code word **c**, although the code word b1, b2, or **c** is modulated into another type of code word according to a boundary rule, arranging the code streams Y1 and Y2 to have opposite parameters INV.